

Плагин для Moodle

Лайки и дизлайки

Тестирования

1. Модульные тесты были выполнены путем написания программы (код программы в приложении), которая создаёт запросы и сравнивает вернувшийся ответ с ожидаемым. Тестируемые запросы:

- Установка лайка
- Установка дизлайка
- Удаление оценки
- Включение/выключение отображения блока
- Включение блока во всех активностях курса
- Выключение блока во всех активностях курса

Результаты модульного тестирования:

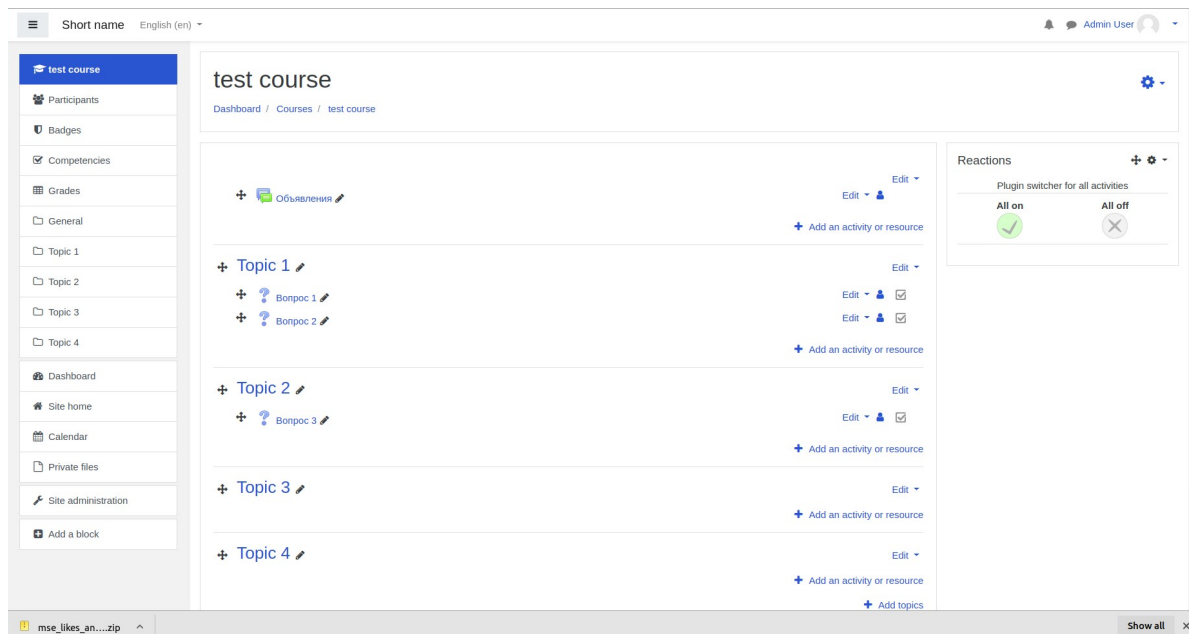
```
Reactions requests
✓ Set like (56ms)
✓ Set dislike
✓ Unset reaction

Settings request
✓ Toggle visibility (41ms)
✓ Toggle visibility 2 (57ms)
✓ Enable course reactions (38ms)
✓ Disable course reactions

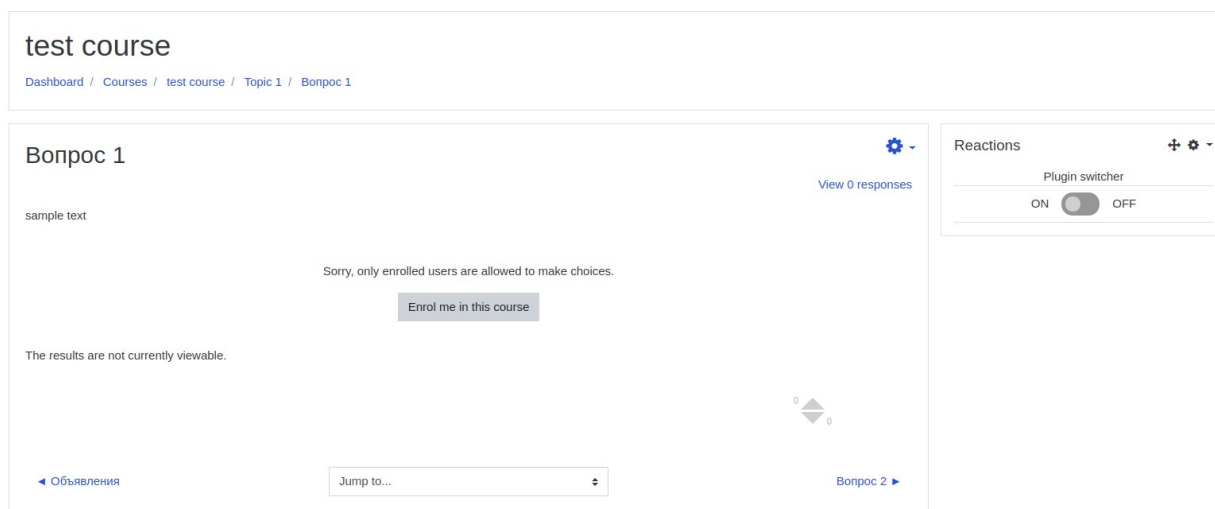
7 passing (289ms)
```

2. и тестирование:

- Включение блока реакций во всех активностях курса:



Блок появился во всех активностях курса:



test course

[Dashboard](#) / [Courses](#) / [test course](#) / [Topic 1](#) / [Вопрос 2](#)

Вопрос 2

⚙

View 0 responses

Вопрос

Sorry, only enrolled users are allowed to make choices.

Enrol me in this course

The results are not currently viewable.

0

0

◀ Вопрос 1

Jump to...

Вопрос 3 ▶

Reactions

+

⚙

◻

Plugin switcher

ON ☒ OFF

test course

[Dashboard](#) / [Courses](#) / [test course](#) / [Topic 2](#) / [Вопрос 3](#)

Вопрос 3

⚙

View 0 responses

Вопрос

Sorry, only enrolled users are allowed to make choices.

Enrol me in this course

The results are not currently viewable.

0

0

◀ Вопрос 2

Jump to...

Reactions

+

⚙

◻

Plugin switcher

ON ☒ OFF

- Выключение блока реакций для активности:

test course

[Dashboard](#) / [Courses](#) / [test course](#) / [Topic 2](#) / [Вопрос 3](#)

Вопрос 3

⚙

View 0 responses

Вопрос

Sorry, only enrolled users are allowed to make choices.

Enrol me in this course

The results are not currently viewable.

◀ Вопрос 2

Jump to...

Reactions

+

⚙

◻

Plugin switcher

ON ☐ OFF

- Выставление реакции:

test course

[Dashboard](#) / [Courses](#) / [test course](#) / [Topic 1](#) / [Вопрос 1](#)

Вопрос 1



[View 0 responses](#)

sample text

Sorry, only enrolled users are allowed to make choices.

[Enrol me in this course](#)

The results are not currently viewable.



[← Объявления](#)

Jump to...



[Вопрос 2 →](#)

Reactions



Plugin switcher

ON ☒ OFF

ПРИЛОЖЕНИЕ

Код программы тестирования

```
const MOODLE_ROOT = 'http://localhost:1010'
const TOKEN = '3e0d77612cd6ed7f9db4671a5af6c006'
const COURSE = 2
const MODULE = 2

const assert = require('assert')
const request = require('supertest')(MOODLE_ROOT)

function get_reaction() {
  return request
    .post('/webservice/rest/server.php')
    .type('form')
    .send({
      'wstoken': TOKEN,
      'wsfunction': 'mse_ld_get_reaction',
      'moduleid': MODULE,
      'moodlewsrestformat': 'json'
    })
}

function set_reaction(reaction) {
  return request
    .post('/webservice/rest/server.php')
    .type('form')
    .send({
      'wstoken': TOKEN,
      'wsfunction': 'mse_ld_set_reaction',
      'moduleid': MODULE,
      'reaction': reaction,
      'moodlewsrestformat': 'json'
    })
}

describe('Reactions requests', function() {

  it('Set like', (done) => {
    set_reaction(1)
    .expect(200)
    .then(() => {
      get_reaction()
        .expect(200)
        .then((res) => {
          assert.equal(res.body, 1)
          done()
        })
    })
  })
})
```

```

    })

    it('Set dislike', (done) => {
        set_reaction(0)
        .expect(200)
        .then(() => {
            get_reaction()
            .expect(200)
            .then((res) => {
                assert.equal(res.body, 0)
                done()
            })
        })
    })

    it('Unset reaction', (done) => {
        set_reaction(2)
        .expect(200)
        .then(() => {
            get_reaction()
            .expect(200)
            .then((res) => {
                assert.equal(res.body, 2)
                done()
            })
        })
    })
});

function get_visibility() {
    return request
        .post('/webservice/rest/server.php')
        .type('form')
        .send({
            'wstoken': TOKEN,
            'wsfunction': 'mse_ld_get_module_reactions_visibility',
            'moduleid': MODULE,
            'moodlewsrestformat': 'json'
        })
}

function toggle_visibility() {
    return request
        .post('/webservice/rest/server.php')
        .type('form')
        .send({
            'wstoken': TOKEN,
            'wsfunction': 'mse_ld_toggle_module_reaction_visibility',
            'moduleid': MODULE,
            'moodlewsrestformat': 'json'
        })
}

```

```

    })
}

function set_course_reactions_visible(visible) {
    return request
        .post('/webservice/rest/server.php')
        .type('form')
        .send({
            'wstoken': TOKEN,
            'wsfunction': 'mse_ld_set_course_modules_reactions_visible',
            'courseid': COURSE,
            'visible': visible,
            'moodlewsrestformat': 'json'
        })
}

```

```

describe('Settings request', function () {

    it('Toggle visibility', (done) => {
        get_visibility()
        .expect(200)
        .then((res) => {
            beginVisibility = res.body
            toggle_visibility()
            .expect(200)
            .then(() => {
                get_visibility()
                .expect(200)
                .then((res) => {
                    assert.equal(res.body, !beginVisibility)
                    done()
                })
            })
        })
    })
})

```

```

it('Toggle visibility 2', (done) => {
    get_visibility()
    .expect(200)
    .then((res) => {
        beginVisibility = res.body
        toggle_visibility()
        .expect(200)
        .then(() => {
            get_visibility()
            .expect(200)
            .then((res) => {
                assert.equal(res.body, !beginVisibility)
                done()
            })
        })
    })
})

```

```

        })
    })
})

it('Enable course reactions', (done) => {
    set_course_reactions_visible(1)
    .expect(200)
    .then(() => {
        get_visibility()
        .expect(200)
        .then((res) => {
            assert.equal(res.body, true)
            done()
        })
    })
})

it('Disable course reactions', (done) => {
    set_course_reactions_visible(0)
    .expect(200)
    .then(() => {
        get_visibility()
        .expect(200)
        .then((res) => {
            assert.equal(res.body, false)
            done()
        })
    })
})

})

```